

一种拟态网络拓扑变换的方法

赵 靛 张校辉 邹 宏 张 鹏

(国家数字交换系统工程技术研究中心 郑州 450002)

摘 要 针对拟态网络构建要求,提出了一种拓扑等效变换的方法。为保证拓扑变换的等效性,提出了一种网络抽象描述方法;基于该描述方法提出基于网络局部等效变换的拟态网络拓扑变换基本思路,并对拟态网络拓扑变换进行了数学建模和理论分析。为了对该方法进行具体说明,设计了基于可用资源的简单等效子网构建算法,并给出了算法的实现过程。最后,分析了该方法的特点,并给出下一步的研究方向。

关键词 拟态网络,拓扑变换,等效变换

中图法分类号 TP393 **文献标识码** A

Method of Topology Transformation for Mimic Network

ZHAO Liang ZHANG Xiao-hui ZOU Hong ZHANG Peng

(National Digital Switching System Engineering and Technology Research Center, Zhengzhou 450002, China)

Abstract According to mimic network construction requirements, we proposed a method of topological equivalent transformation. In order to ensure the equivalence of topology transformation, we proposed a method for describing network abstractly. According to the method for describing network, the basic idea of topology transformation for mimic network based on the local network equivalent transformation was proposed, and mathematical modeling and theoretical analysis for mimic network topological transformation were made. In order to specify the method, this paper designed the algorithm of constructing simple equivalent subnet based on the available resources, and presented implementation of the algorithm. Finally, the analysis of the characteristics of the method, and future research directions were given.

Keywords Mimic network, Topology transformation, Equivalent transformation

1 引言

拟态安全防御的基本思想是在功能等价条件下,以提供目标环境的动态性、非确定性、异构性、非持续性为目的,动态地构建多样化的拟态环境,以防御者可控的方式在多样化环境间实施主动跳变或快速迁移,对攻击者则表现为难以观察和预测的目标环境变化,从而增大包括未知的可利用漏洞和后门在内的攻击难度和代价,即以不确定性对抗未知威胁。基于上述基本思想,提出拟态网络的概念,主要是提供一种可控的方式,使网络按照网络构建者意图在多样化、不确定的目标环境间主动变换、快速迁移,从而提高网络整体安全性。

拟态网络的安全性主要是通过其多样性、不确定性体现出来的,其中拓扑结构的多样性、不确定性是其重要内容。但是拟态网络的拓扑变换除了提供多样性与不确定性以外,一个重要的要求是必须保证网络等效,即拓扑变换后网络的功能、性能等各项指标与变换前的网络一致,这也是拟态网络拓扑变换与其他网络拓扑变换的本质区别。

如何在不影响网络功能、性能的前提下让网络结构根据约束条件和管理者意图进行动态变化是一个需要解决的问题。提供一种满足拟态网络要求的拓扑变换方法,首先必

须明确网络等效的概念和判定方法,然后基于该约束条件给出具体的实现方法。本文第 2 节描述网络拓扑变换的相关工作;第 3 节给出一种网络抽象描述方法,用以确保网络拓扑变换的等效性;第 4 节给出一种基于网络描述方法的拓扑变换方法,并对其中的关键过程进行重点描述;第 5 节给出一个拟态网络拓扑变换实例;第 6 节给出相关实验结论;最后针对本文提出的方法及实验数据给出相关结论,并总结全文。

2 相关工作

可变结构网络可以通过网络拓扑结构的动态变化使网络具有动态性、异构性及不确定性,使网络系统从体系结构角度对攻击者而言具有不可预测性,从而极大增加攻击者的攻击难度。

当前国外关于网络结构随机变化方面的研究包括变形网络、自适应计算机网络以及开放流随机主机转换技术等。2012 年 8 月,美陆军授予雷声公司 310 万美元的“限制地方侦查的变形网络设施”(MORPHINATOR)项目^[1],为其研制具有“变形”能力的计算机网络原型机。该项目主要研究在地方无法探测和预知的情况下,网络管理员有目的地对网络、主机和应用程序进行动态调整和配置,从而预防、延迟或制止网

本文受国家重点基础研究发展计划(“973”计划)(2012CB315901, 2012CB315905),“863”计划课题(2015AA016102)资助。

赵 靛(1979—),女,博士,讲师,CCF 会员,主要研究方向为新型网络体系结构、网络安全, E-mail: zhaoliang790505@126.com;张校辉(1979—),男,博士,讲师,主要研究方向为 SDN、网络流量分析。

络攻击。自适应计算机网络研究方面,2012年5月,美国堪萨斯州大学开始为美空军科学研究办公室研究“自适应计算机网络”^[2],重点研究和量化移动目标防御对计算机网络的影响;研究计算机网络通过自动改变自身设置和结构来对抗在线攻击的可行性,并开发有效的分析模型,以确定移动目标防御系统的有效性。开放流随机主机转换(OFRHM)技术是利用开放流研究移动目标防御体系结构,实现IP地址的不可预知性及高速变换,同时保持配置的完整性,并最小化操作管理。研究结果表明 OFRHM 能够有效防御秘密扫描、蠕虫传播以及其他基于扫描的攻击。

我国信息领域关于网络结构变换的研究主要集中在软件定义网络^[3,4](Software Defined Networking, SDN)和可重构网络方面。特别是在可重构网络方面,有很多关于拓扑构建方法的研究,主要集中在两个方面:1)网络构建或重构的原则、策略,其研究成果可用于指导网络构建^[5-7];2)基于某些特定约束条件的网络构建或重构算法,其中有代表性的是基于可用资源设计的各种网络重构算法。如文献[8]提出一种带迁移同时考虑网络均衡的逻辑承载网构建方法;文献[9]给出在给定资源代价的约束下,为实现资源配置效率的最大化而设计的启发式算法;文献[10]基于路径备份策略解决时延敏感类型的承载网可靠性问题,并提出分阶段处理以进一步优化备份资源消耗;文献[12]将社会网络中心度和拓扑势研究引入到可重构网络虚拟资源分配问题中,提出了基于中心度和拓扑势的虚拟网映射算法。

上述研究要么集中在网络结构变换的顶层设计的概念化阶段,在具体实施方面缺乏研究成果;要么拘泥于某一特定限制条件,缺乏普适性。基于此,本文提出一种拟态网络拓扑变换的方法,从而在网络拓扑变换方面提供系统的解决方案。

3 网络描述

传统的网络描述方法是通过构成网络的节点集合及节点间的链接关系进行描述的,该方法能够准确描述网络拓扑构成,但是在描述网络等效性方面能力不足。为此本文提出一种网络抽象描述方法,该方法不关心网络的具体拓扑实现细节,将其拓扑构成进行屏蔽,而通过网络对外连接点及连接属性等外围特征对网络进行描述。在具体进行网络的数学描述之前,首先定义几个基本概念:

定义 1(节点的邻点集) 用加权图 $G=(V, E)$ 表示网络 G ,如果 $v \in V$,则 $\Gamma(v)=\{u | (u, v) \in E\}$ 称为节点 v 的邻接节点集,简称邻点集。

定义 2(子网到节点的链路) 如果 $G'=(V', E')$ 是加权图 $G=(V, E)$ 中的一个加权子图,即 G' 是 G 的子网,若 $u \in V$ 但 $u \notin V'$,且 $v \in V'$,则 $(u, v) \in E$ 称为子网 G' 到节点 u 的链路,表示为 (u, G') , $b_{G' \rightarrow u}$ 表示子网 G' 到节点 u 的链路带宽, $F_{G' \rightarrow u}$ 表示子网 G' 在该链路上的功能,节点 u 称为子网 G' 的邻接点,简称子网 G' 的邻点。

定义 3(子网的邻点集) 设 $G'=(V', E')$ 是加权图 $G=(V, E)$ 中的一个子图,则 $\Gamma(G')=\{u | (u, G') \in E, u \in V, u \notin V'\}$ 称为子网 G' 的邻点集。

定义 4(子网邻接属性) 用 $G_1=(V_1, E_1)$ 表示子网 G_1 ,若 $A \in \Gamma(G_1)$,用 $l_{G_1}(A)=(F_{G_1 \rightarrow A}, b_{G_1 \rightarrow A})$ 表示子网 G_1 在链

路 (A, G_1) 上的连接属性,由二元组表示,其中第一个元素 $F_{G_1 \rightarrow A}$ 表示链路 (A, G_1) 上实现的功能集,第二个元素 $b_{G_1 \rightarrow A}$ 表示链路 (A, G_1) 的带宽。

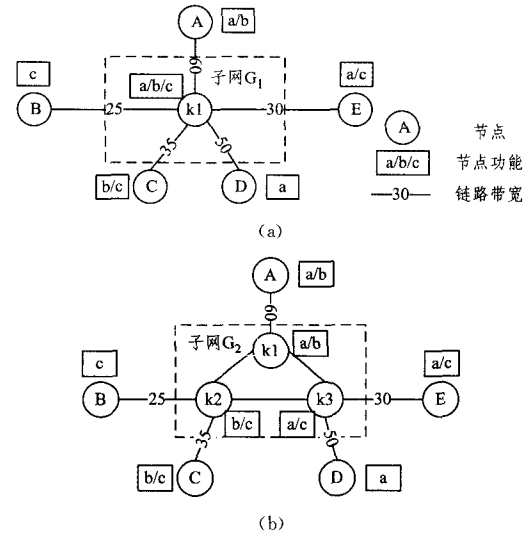


图1 子网描述

按照上述定义,图1所示的两个虚线框内的子网邻点集分别为: $\Gamma(G_1)=\{A, B, C, D, E\}$, $\Gamma(G_2)=\{A, B, C, D, E\}$;两个子网的邻点连接属性分别如表1所列。

表1 子网 G_1 和 G_2 邻接属性

子网 G_1	子网 G_2
$l_{G_1}(A)=(a/b/c, 60)$	$l_{G_2}(A)=(a/b, 60)$
$l_{G_1}(B)=(a/b/c, 25)$	$l_{G_2}(B)=(b/c, 25)$
$l_{G_1}(C)=(a/b/c, 35)$	$l_{G_2}(C)=(b/c, 35)$
$l_{G_1}(D)=(a/b/c, 50)$	$l_{G_2}(D)=(a/c, 50)$
$l_{G_1}(E)=(a/b/c, 30)$	$l_{G_2}(E)=(a/c, 30)$

从子网 G_1 和子网 G_2 的邻接属性描述来看,两个子网对同一邻接点提供的功能属性不同,但是如果再仔细分析就会发现结果并不是这样。因为对于一个子网来说,其功能在通信过程中才能体现出来。举例来说,如果一个子网能够提供功能 a ,但是与其邻接的节点都不具有功能 a ,那么在实际通信过程中该子网的 a 功能无法得以体现。如子网 G_1 虽然能够实现 $a/b/c$ 3 种功能,但是由于其邻接点 A 只具有功能 a/b ,因此子网 G_1 在与邻接点 A 连接的链路上实际也只实现了功能 a/b 。由此,容易发现表1对于子网 G_1 和子网 G_2 的邻接属性描述不准确,子网 G_1 和子网 G_2 的邻接属性分别修正如表2所列。

表2 子网 G_1 和 G_2 邻接属性修正结果

子网 G_1	子网 G_2
$l_{G_1}(A)=(a/b, 60)$	$l_{G_2}(A)=(a/b, 60)$
$l_{G_1}(B)=(c, 25)$	$l_{G_2}(B)=(c, 25)$
$l_{G_1}(C)=(b/c, 35)$	$l_{G_2}(C)=(b/c, 35)$
$l_{G_1}(D)=(a, 50)$	$l_{G_2}(D)=(a, 50)$
$l_{G_1}(E)=(a/c, 30)$	$l_{G_2}(E)=(a/c, 30)$

基于上述修正结果,容易发现子网 G_1 和 G_2 虽然拓扑结构不同,但是用邻接属性描述出来却是相同的,即子网 G_1 和 G_2 对上层用户来说呈现出相同的功能和性能,因此可以认为子网 G_1 和 G_2 是等效的。判定两个子网是否是等效子网,就是判断这两个子网是否能用相同的邻点集及相同的邻接属性描述,而子网的邻点集及邻接属性与子网内部的拓扑结构、子

网内部实现方式等因素无关,只与其外部特征相关,因此提出一种仅用于描述子网外部特征的黑箱子网抽象描述方法。

定义 5(黑箱子网) 用二元组 $B_G = (\Gamma(G), L(G))$ 表示一个黑箱子网 G ,其中 $\Gamma(G)$ 表示该子网的邻点集, $L(G)$ 表示该子网相对于各邻接点的邻接属性集。

黑箱子网的核心就是屏蔽网络的拓扑实现,仅关注其外围属性,从而判定各子网间的关系。如图 1 中的两个子网按照上述定义可描述为如图 2 所示的黑箱子网 G ,即子网 G_1 和 G_2 可以描述为同一个黑箱子网,使用该方法描述的子网不需要关心其内部拓扑结构如何。

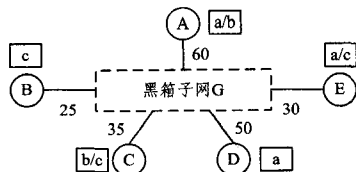


图 2 子网抽象描述方法

黑箱中无论是怎样的拓扑结构,如果对外总能呈现出相同的属性集,则称其为互为等效的子网,等效子网之间可以互相替换而不影响整个网络的功能和性能。由此给出子网等效变换的判定条件:

如果变换前后的子网能够用相同的黑箱子网描述,则这两个子网互为等效子网,所做的变换为等效变换。

该判定条件容易证明得到,这正好满足拟态网络拓扑变换的要求,即网络变换前后的拓扑不同,而功能、性能保持不变,因此可基于该网络描述方法对拟态网络进行拓扑变换,下文主要围绕如何在变换过程中应用该判定条件展开研究。

4 拟态网络拓扑变换模型

基于上一节的网络描述方法,网络中的任何一部分都可以从逻辑上抽象为一个黑箱子网。所谓的网络任何一部分可以是网络中的任意节点、任意链路或者网络中的任何一个联通子网。其中,链路可以描述为具有两个邻接点且其邻接功能属性为透传连接的黑箱子网;单个节点可以描述为具有特定的邻点集,且具有特定邻接属性的黑箱子网,如图 3 所示。

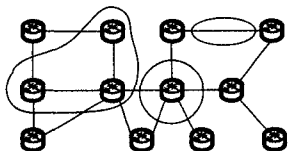


图 3 网络的划分与描述

基于上述分析,网络可以描述为是由若干具有特定属性的黑箱子网组成的,如果对上述所有子网分别进行等效变换,则网络拓扑的局部等效性可以保证拟态网络整体的拓扑变换的等效性。

由此,提出如图 4 所示的拟态网络拓扑变换,其思想为:依次对网络中局部联通子网进行等效变换,通过改变网络的局部拓扑结构,从而达到整体拟态网络拓扑变换的效果。上述思想可以概括总结为两个关键步骤:

A) 确定并描述要进行拓扑等效变换的局部子网,即子网选举;

B) 基于约束条件构建所述子网的等效子网并替换原有子网,即拓扑变换。

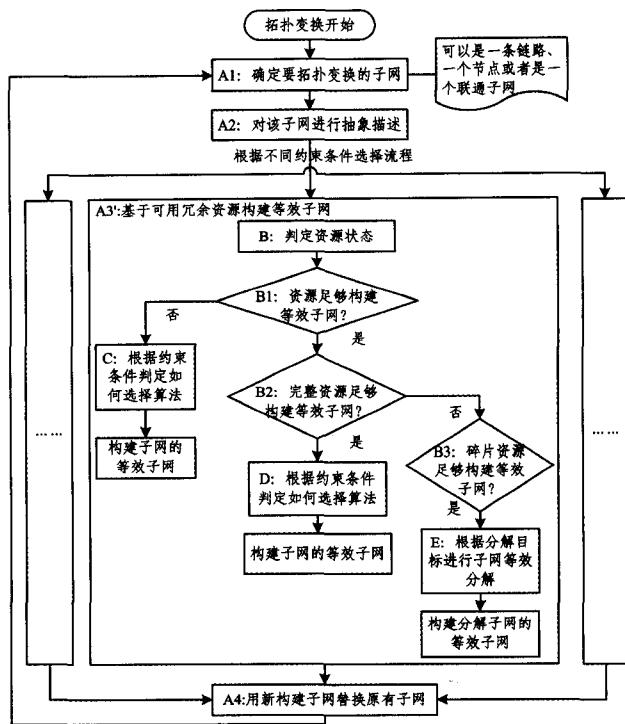


图 4 拟态网络拓扑变换过程

4.1 子网选举模型

基于上述分析,A)是在整个拟态网络中确定一个要进行拓扑变换的子网,该子网的规模和位置可以根据管理决策者意图任意定义,只要能网络抽象描述方法描述即可,即要进行拓扑变换的子网可以是整个网络,也可以是网络中的一部分,如一个节点、一条链路或一个联通子网。而在具体实现过程中很难确定如何从网络的哪一个局部开始进行拓扑变换,因此还需要特定的约束条件或推选策略用以确定需要进行拓扑等效变换的局部子网。

用加权网 $G=(V, E)$ 表示拟态网络的拓扑结构;用 G_s 表示要进行拓扑变换的子网, f 表示推举要进行拓扑变换子网的约束条件或推举策略。由此,给出子网选举模型如式(1)所示:

$$f(G) = B_{G_s} \quad (1)$$

其中,输入是整个拟态网络 G ,输出是推举出来要进行拓扑变换的子网 G_s ,表示为黑箱子网 B_{G_s} 。由式(1)可知,对于相同的输入但作用不同的约束条件或推举策略, $f(G)$ 可能得到不同的黑箱子网。这不属于本文研究的内容,本文后续内容主要讨论如何对推举出来的子网进行等效变换。

4.2 拓扑变换模型

基于第 3 节的描述,容易得出基于同一个黑箱子网可以衍生出多种异构的等效子网,即有无穷多个拓扑异构的解。如果不考虑其他任何约束条件,对于推举出来的 B_{G_s} ,理论上存在无穷多个等效子网,这些等效子网都是网络拓扑等效变换的解,并构成所描述子网的拓扑等效变换解空间。为了在上述解空间中确定唯一解,需要设置一些约束条件。

拓扑变换请求:定义对推举出来的黑箱子网 B_{G_s} 的拓扑变换请求为 $R_{ct} = (B_{G_s}, c)$,其中 $B_{G_s} = (\Gamma(G_s), L(G_s))$ 给出了要构造的子网特征, $c \in CS$ 表示对所要构建子网的约束条件,其中 CS (Condition Set)是等效子网构建的约束条件集。对于不同约束条件,基于同一黑箱子网构建出来的等效子网也不同。可选的约束条件包括资源约束条件、拓扑差异性要求等。本文后续内容主要围绕可用冗余资源这一约束条件进行讨论。

可用冗余资源是为了拟态网络进行拟态变换预设的资源,这些资源可以根据需要进行重构变换,以适应不同需求。冗余资源包括:网络交换节点及用于连接网络节点的链路等;相应的冗余资源状态包括网络交换节点状态和链路状态。对于网络交换节点,其状态包括可用节点的分布、各节点功能;对于链路,其状态主要是指可用链路带宽。所述的交换节点分布是指网络中的交换节点与其他交换节点的关系,如果某两个交换节点中间有链路相连,则该两个节点具有邻接关系,交换节点分布则由该交换节点与哪些节点具有邻接关系来描述。所述的交换节点功能是指交换节点能够交换的数据类型,具体包括:IPv4、IPv6、MPLS、OpenFlow等。

用加权网 $G=(V, E)$ 表示拟态网络中的资源分布,已占用的资源用 $G_{nn}=(V_{nn}, E_{nn})$ 表示,可用冗余资源用 $G_{dle}=(V_{dle}, E_{dle})$ 表示,这即是可用于进行拓扑变换的资源。在不同时刻,可用冗余资源状态不同,而资源状态影响等效子网的构建。由此,若拓扑变换请求 $R_{-t}=(B_{G_s}, c)$ 中的约束条件 c 是资源约束条件,则其拓扑变换方案 f_c 可以描述为从黑箱子网 B_{G_s} 到 G_{dle} 的子图 G_s' 的函数关系如下:

$$f_c(B_{G_s}, G_{dle}) = G_s' \quad (2)$$

本文后续部分主要研究基于可用资源 G_{dle} 如何构建黑箱子网 B_{G_s} 的异构等效子网 G_s' ,即如何实现函数 f_c 。

5 基于可用资源的等效子网构建算法

资源状态影响等效子网构建算法的选择与应用,即使在

$$B = \begin{bmatrix} a/b/c & 0 & 0 & 1000 & 800 & 0 & 0 & 0 & 0 & 0 \\ 0 & b/c/d & 0 & 1000 & 1000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a/b/c/d & 1000 & 1000 & 0 & 0 & 0 & 0 & 0 \\ 1000 & 1000 & 1000 & a/b/c & 0 & 700 & 1000 & 0 & 0 & 0 \\ 800 & 1000 & 1000 & 0 & a/b/c/d & 1000 & 1000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 700 & 1000 & a/c/d & 0 & 900 & 0 & 1200 \\ 0 & 0 & 0 & 1000 & 1000 & 0 & a/b/c/d & 800 & 0 & 1000 \\ 0 & 0 & 0 & 0 & 0 & 900 & 800 & b/c/d & 900 & 800 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 900 & b/c & 0 \\ 0 & 0 & 0 & 0 & 0 & 1200 & 1000 & 800 & 0 & a/b/c \end{bmatrix} \quad (4)$$

上述描述方法仅能描述邻接点的相邻关系,对于节点间的可达性,则描述不足。为了描述网络中节点之间的可达性,用可达加权网和可达路径描述。首先定义如下概念。

定义 6(路径带宽) 节点之间有路径可达,该路径所经过的链路中链路带宽最小值即为这两个节点之间的路径带宽。两个节点之间可能有多条路径可达,因此也会有多个路径带宽。

定义 7(可达加权网) 在加权网 $G=(V, E)$ 中,若节点 v_i 和 v_k 间有路径可达,且路径带宽为 a_{ik} ;同时 v_k 和 v_j 间也有路径可达,且路径带宽为 a_{kj} ;则节点 v_i 和 v_j 间有路径可达,且节点 v_i 和 v_j 间的路径带宽为 a_{ik} 和 a_{kj} 中的小者,则把节点 v_i 和 v_j 间的加权路径加入到加权网 G 中。将符合这种传递关系的附加加权路径都添加到网 G 中,得到的网 G' 即是加权网 G 的可达加权网。

如果加权网中的节点之间有多条路径可达,则都选择具有最大路径带宽的加权路径添加到网 G 中,得到的可达加权网称为最大可达加权网,记为 G'_{max} ;相对地,如果在多条可达路径中都选择具有最小路径带宽的加权路径添加到网 G 中,得到的可达加权网就称为最小可达加权网,记为 G'_{min} 。用一

相同资源状态条件下也会因为构建算法的不同而得到异构的等效子网,因此首先需要对可用资源进行判定,进而再选择适用的算法。为了方便描述,首先对网络资源进行描述如下。

5.1 网络资源描述

用加权网 $G=(V, E)$ 表示网络资源,其中 V 是非空有限节点集,表示网络资源中的节点集, $|V|=n$ 表示节点个数;在网络中节点是有功能属性的,用 F_i 表示节点 i 的功能集合; E 是有限加权链路集, $(i, j) \in E$ 表示节点 i 和 j 是邻接节点, b_{ij} 表示这两个节点之间的带宽。规定网络系统中不存在节点自己到自己的链路,用 $n \times n$ 的加权邻接矩阵 B 表示网络资源为:

$$B[i, j] = \begin{cases} b_{ij}, & \text{若 } (i, j) \in E \\ 0, & \text{若 } (i, j) \notin E \\ F_i, & \text{若 } i=j \end{cases} \quad (3)$$

假设网络资源如图 5 所示。

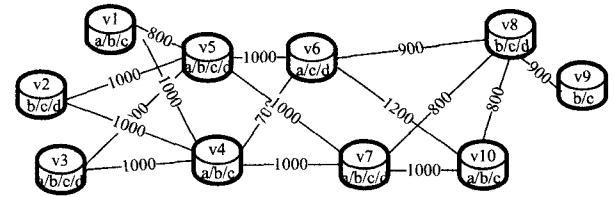


图 5 网络资源 G

图 5 所示的网络资源表示为邻接矩阵 B ,如式(4)所示:

个 $n \times n$ 的加权矩阵 A 表示 G' ,其中 a_{ij} 表示从节点 v_i 到 v_j 的路径带宽, F_i 是节点 i 的功能集合,则:

$$A[i, j] = \begin{cases} a_{ij}, & \text{若 } i \text{ 到 } j \text{ 有路径可达} \\ 0, & \text{若 } i \text{ 到 } j \text{ 没有路径可达} \\ F_i, & \text{若 } i=j \end{cases} \quad (5)$$

可达加权网可以用来描述网络中节点和节点之间的可达性,但是还无法描述节点之间的可达路径,这里用矩阵来描述网络中节点之间的可达路径,称为路径矩阵,用矩阵 C 来表示。若节点 v_i 到 v_j 有可达路径,且路径经过中间节点,则用 c_{ij} 表示节点 v_i 出发到 v_j 经过的第一个节点,即:

$$C[i, j] = \begin{cases} c_{ij}, & c_{ij} \text{ 是 } v_i \text{ 到 } v_j \text{ 的可达路径中的第一个节点} \\ 0, & \text{若 } i \text{ 到 } j \text{ 没有路径可达} \end{cases} \quad (6)$$

若给定网络 G ,求解其最大可达加权图 G'_{max} 和路径矩阵 C 的直观算法如算法 1 所示。

算法 1 求解可达加权图和路径矩阵

Input: 加权图 G 的加权邻接矩阵 $B_{n \times n}$

Output: 最大可达加权图 G'_{max} 的可达加权矩阵 $A_{n \times n}$ 和路径矩阵 $C_{n \times n}$

1. $A \leftarrow B$ and 初始化 C ; // 将矩阵 B 复制到矩阵 A

2. 重复以下过程,直到 A 的各项在一次完整的三重循环中不再改变;
3. for $i \leftarrow 1$ to n do
4. for $j \leftarrow 1$ to n do
5. for $k \leftarrow 1$ to n do
6. if $a_{ik} \neq 0$ and $a_{kj} \neq 0$ then $a'_{ij} \leftarrow \min(a_{ik}, a_{kj})$;
7. if $a'_{ij} > a_{ij}$ then $a_{ij} \leftarrow a'_{ij}$ and $c_{ij} = k$;

$$A = \begin{bmatrix} a/b/c & 1000 & 1000 & 1000 & 800 & 800 & 1000 & 800 & 800 & 1000 \\ 1000 & b/c/d & 1000 & 1000 & 1000 & 1000 & 1000 & 900 & 900 & 1000 \\ 1000 & 1000 & a/b/c/d & 1000 & 1000 & 1000 & 1000 & 900 & 900 & 1000 \\ 1000 & 1000 & 1000 & a/b/c & 1000 & 700 & 1000 & 900 & 900 & 1000 \\ 800 & 1000 & 1000 & 1000 & a/b/c/d & 1000 & 1000 & 900 & 900 & 1000 \\ 8000 & 1000 & 1000 & 700 & 1000 & a/c/d & 1000 & 900 & 900 & 1200 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & a/b/c/d & 800 & 900 & 1000 \\ 800 & 900 & 900 & 900 & 900 & 900 & 800 & b/c/d & 900 & 800 \\ 800 & 900 & 900 & 900 & 900 & 900 & 900 & 900 & b/c & 900 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1200 & 1000 & 800 & 900 & a/b/c \end{bmatrix}$$

$$C = \begin{bmatrix} v1 & v4 & v4 & v4 & v5 & v5 & v4 & v4 & v4 & v4 \\ v4 & v2 & v4 & v4 & v5 & v5 & v4 & v5 & v5 & v4 \\ v4 & v4 & v3 & v4 & v5 & v5 & v4 & v4 & v4 & v4 \\ v1 & v2 & v3 & v4 & v2 & v6 & v7 & v7 & v7 & v7 \\ v1 & v2 & v3 & v2 & v5 & v6 & v7 & v6 & v6 & v6 \\ v5 & v5 & v5 & v5 & v5 & v6 & v5 & v8 & v8 & v10 \\ v4 & v4 & v4 & v4 & v5 & v5 & v7 & v5 & v5 & v10 \\ v6 & v6 & v6 & v6 & v6 & v6 & v6 & v8 & v9 & v6 \\ v8 & v8 & v8 & v8 & v8 & v8 & v8 & v8 & v9 & v8 \\ v6 & v6 & v6 & v7 & v6 & v6 & v7 & v6 & v6 & v10 \end{bmatrix}$$

5.2 资源绑定条件

基于 5.1 节的资源描述方法可对网络冗余资源做基本判定,以初步确定可适用的等效子网构建算法。本节主要对冗余资源总量进行判定,以确定冗余资源是否足够用以构建等效子网。

资源判定条件 1 若黑箱子网 B_G 的邻点集中存在一个节点,该节点的邻点集的功能集合小于黑箱子网在该邻接点的功能需求,或该节点到邻点集中所有节点的带宽之和小于黑箱子网在该邻接点的带宽需求,则该黑箱子网的等效子网不存在。

反证:

(1) 证明基于功能集合的判定条件

黑箱子网 B_G 的邻点集 $\Gamma(G)$ 中存在一个节点 v_j , 即 $v_j \in \Gamma(G)$, 若该节点的邻点集 $\Gamma(v_j) = \{v_1, \dots, v_m\}$, 则有:

$$F_{v_j} = \bigcup_{i=1}^m F_{v_i} \subset F_{G \rightarrow v_j} \quad (7)$$

假设能够成功构建 B_G 的等效子网,则根据等效子网的构建原理,等效子网在邻接点 v_j 的功能集应满足:

$$F_{G \rightarrow v_j} \subseteq F_{v_j} \quad (8)$$

显然式(7)和式(8)矛盾,因此假设错误。

(2) 同理证得基于带宽的判定条件亦成立。证毕。

资源判定条件 1 给出了最基本的资源约束,如果网络冗余资源足够构建相应的等效子网,则可以继续后续算法,否则无论选择任何算法都无法得到需要的结果。同时,该判定条件只是构建等效子网的必要条件,而非充分条件,因此无法保证满足资源判定条件 1 的情况下一定能够成功构建等效子网,也无法明确如何选择最适合的等效子网构建算法。

8. end

9. end

10. end

算法的最坏时间复杂度为 $O(n^3 \log n)$ 。

图 5 的网 G 依据算法 1 得到的最大可达加权矩阵 A 和最大路径矩阵 C 分别为:

5.3 简单等效子网构建算法

在相同资源状态下,可能存在同一黑箱子网的多个拓扑异构解,这可以通过不同的构建算法得到,因此选择什么算法直接影响最后的结果。本节讨论简单等效子网构建算法 (Simple Equivalent Sub-network Constructing Algorithm, SESCOA),用以构建拓扑最简单的等效子网,即拓扑所用到的节点和链路资源最少的等效子网。

为了方便后续描述,首先给出几个定义:

定义 8(邻接备选节点) 如果节点 v_i 与黑箱子网 B_G 的邻点集 $\Gamma(G)$ 中的某个节点 v_j 相邻,且其邻接属性 $l_{v_i}(v_j) \geq l_G(v_j)$, 则节点 v_i 称为 v_j 相对于子网 B_G 的邻接备选节点,简称 v_j 的备选节点。 v_j 所有备选节点组成的集合称为 v_j 的备选节点集,记为 $S_{v_j \rightarrow G}$ 。

定义 9(节点复用度) 如果节点 v_i 是黑箱子网 B_G 的邻点集 $\Gamma(G)$ 中多个节点的邻接备选节点,则称节点 v_i 为复用备选节点,简称为复用节点,而节点 v_i 作为邻接备选节点的次数就称为节点复用度。

对于邻点集只有两个节点的黑箱子网,其最简单的等效子网即是一条能够替代原有子网的等效链路。假设黑箱子网 B_G 的邻点集 $\Gamma(G) = \{v_i, v_j\}$, 用 $e = (v_i, v_j)$ 表示节点 v_i 和 v_j 之间的链路, b_{ij} 表示节点 v_i 和 v_j 之间链路的带宽。如果冗余资源中存在 $e = (v_i, v_j)$, 且 $b_{ij} > b_{G \rightarrow v_i}$ 且 $b_{ij} > b_{G \rightarrow v_j}$, 则可以很容易找到等效链路。对于邻点集有多个节点的黑箱子网,最简单的等效子网是一个与所有邻点集中所有邻接点都有链路连接的单节点,且该单节点的功能集和各个链路带宽都大于等于黑箱子网要求的邻接属性。

如果冗余资源中不存在上述单点资源,则需要进一步构建多节点等效子网。假设 SESCOA 算法是在冗余资源充足的条件下运行的,即一定能够基于冗余资源构建出满足需求的等效子网。对于邻点集只有两个节点的黑箱子网 B_G , 要构建的等效子网就是从节点 v_i 到 v_j 的路径 c_{ij} ($c_{ij} \neq 0$), 该路径上的节点具有透传功能,且路径带宽 $a_{ij} \geq \max(b_{G \rightarrow v_i}, b_{G \rightarrow v_j})$ 。为了得到最短路径,可以使用广度优先搜索算法在冗余资源中进行查找。

对于邻点集有多个节点的黑箱子网,从上述选择单点资源的过程可以看出,能够用来替代黑箱子网的单节点的复用度是最大的,等于黑箱子网 B_G 的邻点集 $\Gamma(G)$ 中的节点个

数,即在构建等效子网时使用的节点复用度越大,则所需要的节点越少。按照这个思路在多节点构建方案中首先要选择复用度尽量大的节点参与构建。基于上述思想,我们首先按照节点复用度对备选资源中的可用节点进行排序,同时记录复用节点的邻接性。节点复用度就是节点参与等效子网构建的优先级,越是排在前面的节点说明该节点与更多的邻接节点有满足需求的功能和链接带宽,因此优先级更高。但优先级高的节点是否能够被选中参与等效子网的构建,还需要看该节点与黑箱子网的其他邻接节点的可达性。如果与其他邻接节点不可达,则依次按照优先级顺序选择后面的备选节点并求解其可达路径。

在求解备选节点与黑箱子网的其他邻接节点的可达路径过程中,为了避免下一步构建处理选到已经被上一步构建处理占用的资源,造成构建的等效子网不可用,需要在每一次预选节点和链路之后及时更新可用资源。如果不存在备选节点到其他邻接点的有效路径,则按照备选节点集中节点的优先级顺序依次查找后续节点与其他邻接点的有效路径。基于上述求解方案一定能够找到从备选节点到其他邻接点的路径,从而可以构建出相应的等效子网。

6 SESCO 算法实现

基于 5.3 节的算法描述,给出具体的算法实现过程。假设需要进行拓扑变换的子网为 B_G ,用于构建等效子网的冗余资源 $G_{idle} = (V_{idle}, E_{idle})$,用 $n \times n$ 的加权矩阵 $B_{n \times n}$ 、 $A_{n \times n}$ 和 $C_{n \times n}$ 分别表示其邻接关系、加权可达资源和可达路径; $n \times n$ 的加权矩阵 $B_{n \times n}'$ 表示构造出来的等效子网邻接关系。算法实现过程分为以下几个步骤:

1. 在冗余资源中搜索单链路资源。

PROCEDURE LinkSearch($B_G, B_{n \times n}, B_{n \times n}'$)

1. 提取 $\Gamma(G)$ 中的节点 v_i 和 v_j , 及 $b_{G \rightarrow v_i}$;
2. if $b_{ij} \geq b_{G \rightarrow v_i}$ then do //冗余资源中存在满足需求的链路
3. $b_{ij}' = b_{G \rightarrow v_i}$;
4. $b_{ij} = b_{ij} - b_{G \rightarrow v_i}$;
5. 输出 $B_{n \times n}'$;
6. else 失败返回;
7. end if

2. 在冗余资源中搜索单节点资源。

PROCEDURE NodeSearch($B_G, B_{n \times n}, B_{n \times n}'$)

1. 初始化 B_G 的功能集 F_G ;
2. for $i=1$ to n
3. if $v_i \in \Gamma(G)$ then do
4. $F_{G \rightarrow v_i} \rightarrow F_G$;
5. end if
6. end
7. for $i=1$ to n
8. if $\Gamma(v_i) \supseteq \Gamma(G)$ and $F_i \supseteq F_G$ then do
9. each v_j in $\Gamma(v_i)$
10. if $v_j \in \Gamma(G)$ and $b_{ij} \geq b_{G \rightarrow v_j}$ then do
11. $b_{ij}' = b_{G \rightarrow v_j}$;
12. $b_{ij} = b_{ij} - b_{G \rightarrow v_j}$;
13. end if
14. end
15. 输出 $B_{n \times n}$ 和 $B_{n \times n}'$;
16. end if
17. end

3. 构建等效链路。

PROCEDURE LinkBuild($B_G, B_{n \times n}, A_{n \times n}, C_{n \times n}, B_{n \times n}'$)

1. 提取 $\Gamma(G)$ 中的节点 v_i 和 v_j , 及 $b_{G \rightarrow v_i}$;
2. if $b_{ij} \neq 0$ then do //首选直连链路,不足带宽通过可达路径补充
3. $b_{ij}' = b_{ij}$, $b_{G \rightarrow v_i} = b_{G \rightarrow v_i} - b_{ij}$, $b_{ij} = 0$;
4. for $j=1$ to n
5. if $a_{ij} \geq b_{G \rightarrow v_i}$ then do //冗余资源中存在满足需求的单链路资源
6. $l=i$ and $k=c_{ij}$;
7. while $k \neq j$ do
8. $b_{a-lk} = b_{G \rightarrow v_i}$;
9. $b_{lk} = b_{lk} - b_{G \rightarrow v_i}$;
10. $l=k$ and $k=c_{lj}$;
11. end
12. 输出 $B_{n \times n}$ 和 $B_{n \times n}'$;
13. else 失败返回;
14. end if
15. end
16. end if

4. 在冗余资源中选择备选节点,并基于节点复用度进行排序。用 $P_n = (p_1, \dots, p_n)$ 分别记录各备选节点的复用度, $ID_n = (id_1, \dots, id_n)$ 记录各备选节点的邻接性, V 是排好序的备选节点集,则根据复用度对备选节点进行排序的过程如下:

PROCEDURE StandbyNodeSort($B_G, B_{n \times n}, ID_n, P_n, V$)

1. 初始化 ID_n, P_n 和 V ;
2. each v_j in $\Gamma(G)$
3. for $i=1$ to n
4. if $F_i \supseteq F_{G \rightarrow v_j}$ and $b_{ij} \geq b_{G \rightarrow v_j}$ then do
5. 在 id_i 中记录与 v_j 相邻;
6. p_i++ ;
7. end if
8. end
9. end
10. 根据 P_n 中记录的各节点的复用度对备选节点排序并放入 V ;
11. 输出 V

5. 查找备选节点到其他邻接点的可达路径。 v_k 是从 V 中选出的节点,其节点复用度 $|v_k| < |\Gamma(G)|$,用 $U = \Gamma(G) - \Gamma_G(v_k)$ 表示 $\Gamma(G)$ 中 v_k 不邻接的节点集。 $n \times n$ 的加权矩阵 $A_{n \times n}$ 表示当前可用资源的加权可达性, $n \times n$ 的路径矩阵 $C_{n \times n}$ 表示节点之间的可达路径。基于资源 $A_{n \times n}$ 求解节点 v_k 到 U 的可达路径也有多种方案,我们给出其中一种求解方案:

PROCEDURE PathSolve($v_k, U, A_{n \times n}, C_{n \times n}, B_{n \times n}'$)

1. 初始化 $B_{n \times n}'$;
2. each v_j in U
3. if $a_{kj} \geq b_{G \rightarrow v_j}$ then do
4. 在 $B_{n \times n}'$ 中记录从 v_k 到 v_j 的路径资源;
5. end if
6. 输出 $B_{n \times n}'$;
7. end

根据上述基本过程,给出 SESCO 算法实现如下:

算法 SESCO:

输入:故障 B_G , 当前冗余资源 $B_{n \times n}$, 加权可达矩阵 $A_{n \times n}$, 路径矩阵 $C_{n \times n}, ID_n, P_n, V$

输出: $B_{n \times n}'$

1. 初始化 $B_{n \times n}'$;
2. if $|\Gamma(G)| \leq 2$ then do

参考文献

- [1] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74
- [2] 左青云, 陈鸣, 赵广松, 等. 基于 OpenFlow 的 SDN 技术研究[J]. 软件学报, 2013, 24(5): 1078-1097
- [3] 李丹, 陈贵海, 任丰原, 等. 数据中心网络的研究进展与趋势[J]. 计算机学报, 2014, 37(2): 259-274
- [4] 窦军, 陈文佳. SUPANET 基 OAM 的保护交换研究[J]. 计算机科学, 2011, 38(4): 87-92
- [5] 窦军. 单层用户数据交换平台体系结构研究[D]. 成都: 西南交通大学, 2011
- [6] 林闯, 贾子骁, 孟坤. 自适应的未来网络体系架构[J]. 计算机学报, 2012, 35(6): 1077-1093
- [7] 戴彬, 王航远, 徐冠, 等. SDN 安全探讨: 机遇与威胁并存[J]. 计算机应用研究, 2014, 31(8): 2254-2262
- [8] 薛聪, 马存庆, 刘宗斌, 等. 一种安全 SDN 控制器架构设计[J]. 信息安全学报, 2014(9): 34-38
- [9] ONF Market Education Committee. Software-Defined Networking: The new norm for networks[EB/OL]. (2012-04-13). <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf>
- [10] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74
- [11] Tootoonchian A, Gorbunov S, Ganjali Y, et al. On controller performance in software-defined networks[C]//USENIX Workshop on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services(Hot-ICE). 2012: 10
- [12] 江国龙, 付斌章, 陈明宇, 等. SDN 控制器的调研和量化分析[J]. 计算机科学与探索, 2014, 8(6): 653-664
- [13] Braga R, Mota E, Passito A. Lightweight DDoS flooding attack detection using NOX/OpenFlow[C]//2010 IEEE 35th Conference on Local Computer Networks(LCN). IEEE, 2010: 408-415
- [14] Wang B, Zheng Y, Lou W, et al. DDoS Attack Protection in the Era of Cloud Computing and Software-Defined Networking[C]//2014 IEEE 22nd International Conference on Network Protocols(ICNP). IEEE, 2014: 624-629
- [15] 陶冶, 张尼, 张云勇, 等. SDN 安全防护技术研究[J]. 电信技术, 2014(6): 14-17
- [16] Jose L, Yu M, Rexford J. Online measurement of large traffic aggregates on commodity switches[C]//Proc. of the USENIX HotICE workshop. 2011: 13-13
- [17] Yao G, Bi J, Xiao P. Source address validation solution with OpenFlow/NOX architecture[C]//2011 19th IEEE International Conference on Network Protocols(ICNP). IEEE, 2011: 7-12
- [18] Dover J M. A denial of service attack against the Open Floodlight SDN controller[EB/OL]. [2013-12-30]. <http://dovernetworks.com/wp-content/uploads/2013/12/OpenFloodlight-1230-2013.pdf>

(上接第 328 页)

3. if 执行 LinkSearch()过程成功 then do;
4. 成功返回;
5. else if 执行 LinkBuild()过程成功;
6. 成功返回;
7. end if
8. else do
9. if 执行 NodeSearch()过程成功 then do;
10. 成功返回;
11. else do
12. 执行 StandbyNodeSort()过程, 得到排序的节点集 V;
13. if 节点集 V 为空集 then do
14. 失败退出;
15. else do
16. 按序从 V 中取节点 v_k ;
17. if 执行 PathSolve()过程成功 then do
18. 成功返回;
19. end if
20. end if
21. end if
22. 失败退出

结束语 对本文提出的拟态网络拓扑等效变换方法总结如下: 1) 通过多次对网络局部子网的拓扑等效变换达到整体网络拓扑变换的效果, 由于拓扑变换的目标是网络中的子网, 而非整个网络, 因此可以减小拓扑变换的网络规模, 降低网络拓扑变换的难度; 2) 基于本文提供的子网描述方法, 可以对网络中的任意部分进行抽象描述, 可以是一条链路、一个节点或一个连通子网, 甚至是整个网络, 从而可以增加网络拓扑变换的灵活性。

本文提出的拟态网络拓扑变换方法只是一些基础性工作, 在此基础上还可以开展更多的工作, 如: 1) 细化资源判定条件, 为算法选择提供理论依据; 2) 丰富算法类型, 为同等条

件下生成异构子网提供更多方法。

参考文献

- [1] Unruh I, Bardas A G, Zhuang Rui, et al. Compiling abstract specifications into concrete systems-bringing order to the cloud[C]//Proceedings of the 28th Large Installation System Administration Conference(LISA14). Seattle, WA, 2014: 17-33
- [2] Deloach S, Ou Xin-ming, Zhuang Rui, et al. Model-driven, moving-target defense for enterprise network security[J]. Lecture Notes in Computer Science, 2014, 8378: 137-161
- [3] 林闯, 贾子骁, 孟坤. 自适应的未来网络体系架构[J]. 计算机学报, 2012, 35(6): 1077-1093
- [4] 王淑玲, 李济汉, 张云勇, 等. SDN 架构及安全性研究[J]. 电信科学, 2013, 3: 117-122
- [5] 刘强, 汪斌强, 徐恪. 基于构件的层次化可重构网络构建及重构方法[J]. 计算机学报, 2010, 33(9): 1557-1568
- [6] 王会勇, 赵海良, 杨晓伟. 基于拓扑变换的一种不确定性推理方法[J]. 重庆交通学院学报(自然科学版), 2004, 23(1): 112-115
- [7] 刘军, 于耕, 张慧鹏. 基于节点控制的空间信息网拓扑重构算法[J]. 电子学报, 2011, 39(8): 1837-1844
- [8] 齐宁, 汪斌强, 郭佳. 逻辑承载网构建方法的研究[J]. 计算机学报, 2010, 33(9): 112-119
- [9] 李黎, 郑庆华, 管晓宏. 基于有限资源提升网络可生存性的拓扑重构方法[J]. 物理学报, 2014, 63(17): 170-201
- [10] 王志明, 汪斌强. 基于备份的可重构服务承载网可靠性映射方法[J]. 电子与信息学报, 2013, 35(1): 126-132
- [11] 李稳国, 邓曙光, 杨冰, 等. 相互依存网络间的拓扑构建方法[J]. 计算机工程与应用, 2014, 50(11): 85-89
- [12] 吕书明, 张明磊, 孙树立. 基于简化和细分技术的三角形网络拓扑优化方法[J]. 计算机辅助设计与图形学报, 2014, 26(8): 156-162
- [13] 王子厚, 韩言妮, 林涛, 等. 可重构网络中基于中心度与拓扑势排序的资源分配算法[J]. 通信学报, 2012, 33(8): 10-20